Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

FreeIPA для Linux-парка: централизованная аутентификация

Как мы собираем 389 DS + Kerberos + Dogtag + SSSD: реплики, HBAC, sudo, PKI и OTP-двухфакторка

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

О ЧЁМ ЭТОТ РАЗБОР

Суть проблемы

Парк Linux-серверов с локальными /etc/passwd и россыпью SSH-ключей: пароли везде разные, увольнение — ручной обход всех машин, sudo раздан целиком через wheel. Мы закрываем это FreeIPA: единый каталог, Kerberos SSO, HBAC и sudo-правила из LDAP. Без этого — «забытые» учётки уволенных, непроходимый аудит ИБ и часы простоя на каждом кадровом событии.

Почему это важно бизнесу

- Учётка уволенного отключается одной командой ipa user-disable за секунды, а не часами обхода серверов
- Аудит ИБ и импортозамещение: доступы централизованы, лицензии Microsoft Server+CAL не нужны
- Наименьшие привилегии: разработчик не попадёт на сервер БД, sudo — только на разрешённые команды
- Свой УЦ: TLS внутренних сервисов продлевается сам — без простоев из-за истёкших сертификатов

ЧТО МЫ ЗАКЛАДЫВАЕМ

Ключевые параметры реализации

4.13.3

актуальная серия FreeIPA (релиз 19.08.2026); сервер — только RHEL-совместимые Rocky/Alma 9, Ub…

freeipa.org

300 с

максимальный дрейф часов для Kerberos (clockskew) — chrony настраиваем до запуска инсталлятора

web.mit.edu/kerberos

28 дней

за столько certmonger сам начинает продление TLS-сертификатов — ручных перевыпусков нет

freeipa.org

x2

минимум серверов: master + реплика через ipa-replica-install, роль CA — на обоих узлах

наш стандарт

≤4

репликационных соглашений на узел; топологию проверяем ipa topologysuffix-verify

docs.redhat.com

14 симв.

минимум пароля админ-групп: pwpolicy --minlength=14 --minclasses=4 --maxfail=3

наш стандарт

РЕАЛИЗАЦИЯ 1 ИЗ 3

Ядро домена: master + реплика с CA на разных гипервизорах

Что настраиваем

Пара VM Rocky Linux 9 (4 vCPU / 8 ГБ / SSD) под 389 DS, Kerberos KDC, Dogtag CA и BIND; с офисом клиента — VPN

Как мы это делаем

1. Готовим узел: hostname -f возвращает FQDN, chronyd синхронизирован, firewall-cmd --permanent --add-service={freeipa-4,dns,ntp} и reload
2. ipa-server-install --setup-dns с форвардерами и --no-reverse в unattended-режиме; пароли Directory Manager и admin — сразу в сейф; ~15 минут на развёртывание
3. Реплика: ipa-replica-install --setup-ca --setup-dns на втором гипервизоре; сегменты репликации проверяем ipa topologysegment-find domain
4. На оба узла — ipa-healthcheck.timer (ежедневный прогон 30+ проверок) и ночной ipa-backup с выгрузкой архива в офсайт

РЕЗУЛЬТАТ

Отказ одного узла не останавливает логины: SSSD на клиентах сам переключается на реплику, а кеш (cache_credentials=True) держит вход даже при полной недоступности домена. Восстановление — ipa-restore из свежего архива.

КЛЮЧЕВОЙ НЮАНС

Роль CA обязательно держим на двух узлах (ipa-ca-install на реплике): потеря единственного ключа Dogtag означает переразвёртывание всего PKI. Контроль — ipa server-role-find по роли CA server.

РЕАЛИЗАЦИЯ 2 ИЗ 3

Подключение парка и матрица доступа HBAC + sudo

Что настраиваем

Все Linux-хосты клиента (Rocky/Alma/Ubuntu/Debian) через Ansible-роль поверх ipa-client-install

Как мы это делаем

1. Раскатываем клиентов плейбуком: ipa-client-install --mkhomedir --enable-dns-updates --unattended; SSSD и krb5.conf настраиваются автоматически
2. Первым делом гасим дефолт: ipa hbacrule-disable allow_all — иначе любой пользователь домена получает шелл на любом хосте
3. Строим матрицу: hostgroup по ролям (web, db, monitoring), hbacrule «группа → hostgroup → sshd/login»; каждое правило прогоняем ipa hbactest до применения
4. sudo из каталога: ipa sudocmd-add на конкретные команды, sudocmdgroup и sudorule на группы хостов — вместо членства всех в wheel

РЕЗУЛЬТАТ

Доступы перестают жить в локальных файлах: увольнение — ipa user-disable, новая роль — членство в группе. Кто куда заходит и что запускает через sudo, виден весь срез из одной консоли — и он же уходит в аудит.

КЛЮЧЕВОЙ НЮАНС

hbactest — обязательный шаг: правило, собранное «на глаз», либо не пускает нужных, либо пускает лишних. Прогоняем матрицу пользователь×хост×сервис до отключения allow_all.

РЕАЛИЗАЦИЯ 3 ИЗ 3

Внутренний PKI на Dogtag и двухфакторка OTP

Что настраиваем

TLS-сертификаты внутренних веб-сервисов и 2FA для админских учёток на всех узлах домена

Как мы это делаем

1. ipa service-add HTTP/host, затем ipa-getcert request с ключом -C (post-save: systemctl reload nginx) — выпуск и продление полностью автоматические
2. Корневой CA приезжает на клиентов сам: ipa-client-install кладёт /etc/ipa/ca.crt и добавляет его в системный trust-bundle
3. 2FA: ipa user-mod --user-auth-type=otp + ipa otptoken-add (TOTP); пользователь привязывает FreeOTP по QR-коду и вводит пароль+код слитно
4. Контроль сроков: ipa-getcert list должен показывать MONITORING; истечения системных сертификатов ловим через ipa-healthcheck

РЕЗУЛЬТАТ

Внутренние сервисы получают валидный TLS без Let's Encrypt и без доступа из интернета — браузеры доверяют по корню домена. Админские входы по SSH и в Web UI закрыты вторым фактором без внешних сервисов.

КЛЮЧЕВОЙ НЮАНС

Сроки системных сертификатов Dogtag держим под мониторингом: истёкший сертификат подсистемы кладёт весь выпуск. Лечится ipa-cert-fix, но до этого лучше не доводить.

ЧТО МЫ ВЫУЧИЛИ НА ПРАКТИКЕ

Подводные камни

- ✕ **Короткое имя вместо FQDN**
  Инсталлятор падает на середине, если hostname -f не возвращает полный FQDN; проверяем имя и запись в /etc/hosts до запуска ipa-server-install

- ✕ **Отключённый SELinux**
  FreeIPA рассчитан на enforcing; в permissive/disabled появляются трудноуловимые сбои. Оставляем enforcing — политики идут из коробки

- ✕ **Забытое правило allow_all**
  Пока оно активно, HBAC не работает: любой пользователь заходит на любой хост. Отключаем сразу после установки, до заведения людей

- ✕ **Один сервер без реплики**
  Кеш SSSD маскирует падение: старые сессии живут, новые логины и смены паролей — нет. Вторая реплика с ролью CA — обязательный минимум

- ✕ **Конфликт с существующим DNS**
  При живом BIND/Windows DNS не отдаём FreeIPA всю зону — делегируем поддомен и настраиваем форвардинг в обе стороны

- ✕ **389 DS на шпиндельном диске**
  Каталог генерирует много мелких IOPS; на HDD растут задержки логина и репликации. Базу кладём на SSD, реплику — на другой гипервизор

- ✕ **Ожидание переноса паролей**
  Хеши из /etc/shadow в FreeIPA не импортируются: закладываем сброс со сменой при первом входе, UID/GID сохраняем ключами --uid/--gidnumber

- ✕ **Бэкап без ключей CA**
  Потеря ключа Dogtag = переразвёртывание PKI. Ночной полный ipa-backup выгружаем в офсайт и раз в квартал проверяем ipa-restore на стенде

УРОВНИ РЕАЛИЗАЦИИ

Как правильно

МИНИМУМ

- Один сервер FreeIPA + chrony на всех узлах, hostname -f = FQDN
- allow_all отключён, базовые группы и HBAC-правила по ролям
- Ночной ipa-backup с копией вне площадки

НОРМАЛЬНО

- Реплика с CA (ipa-replica-install --setup-ca) на отдельном гипервизоре
- sudo-правила из каталога вместо wheel, pwpolicy для админ-групп
- ipa-healthcheck.timer ежедневно + алерты в мониторинг
- Клиенты раскатываются Ansible-ролью, а не руками

ХОРОШО

- OTP для админов, SSH-ключи пользователей — в каталоге (--sshpubkey)
- Сертификаты всех внутренних сервисов — через certmonger
- Hidden replica под бэкапы и тесты восстановления
- Регулярный прогон hbactest по всей матрице ролей

ПРОВЕРЬТЕ У СЕБЯ

Чек-лист самопроверки

- [ ] Есть ли вторая реплика и стоит ли роль CA минимум на двух узлах (ipa server-role-find)?
- [ ] Отключено ли HBAC-правило allow_all и описана ли матрица «кто → куда → каким сервисом»?
- [ ] Синхронизировано ли время на всех узлах (дрейф для Kerberos — не больше 300 секунд)?
- [ ] Делается ли ipa-backup ежедневно и проверялось ли восстановление ipa-restore на стенде?
- [ ] Выведены ли sudo-права из локальных sudoers в правила каталога?
- [ ] Отключается ли учётка уволенного одной командой — вместе с SSH-доступом и сертификатами?
- [ ] Продлеваются ли TLS-сертификаты автоматически (статус MONITORING в ipa-getcert list)?
- [ ] Включена ли OTP-двухфакторка хотя бы для административных учёток?
- [ ] Гоняется ли ipa-healthcheck по расписанию и разбирает ли кто-то его отчёты?

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.

ЧТО МЫ ДЕЛАЕМ

Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Разворачиваем FreeIPA под ключ: master + реплика с CA, DNS, миграция учёток из /etc/passwd и выгрузок 1С:ЗУП
- Подключаем весь Linux-парк (Rocky/Alma/Ubuntu/Debian) Ansible-ролью и строим матрицу HBAC/sudo по ролям
- Настраиваем внутренний PKI на Dogtag: автовыпуск и автопродление TLS для внутренних сервисов
- Включаем OTP-двухфакторку и переводим SSH-доступ на ключи из каталога
- Берём домен на сопровождение: ipa-healthcheck, бэкапы, мониторинг реплик в Zabbix

15+
лет в ИТ-поддержке

50
рабочих мест — наш профиль

МТС
дата-центр, Москва

КОНТАКТЫ

Обсудить вашу задачу

Сайт	**itfresh.ru**
Телефон	**+7 903 729-62-41**
Telegram	**@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.

itfresh.ru

ДОКУМЕНТАЦИЯ И НАШ СТЕК

Техническая база

01 FreeIPA Release Notes 4.13.x (freeipa.org — 4.13.3)

02 FreeIPA Workshop: client enrolment, HBAC, cert management (freeipa.readthedocs.io — 4.x)

03 Certmonger — отслеживание и продление сертификатов (freeipa.org — 2026)

04 RHEL 9: Planning and Installing Identity Management (docs.redhat.com — RHEL 9)

05 SSSD: кеширование и офлайн-аутентификация (sssd.io — 2.x)

06 Наш Ansible-шаблон раскатки ipa-client + матрицы HBAC (itfresh.ru — 2026)

Основано на официальной документации продуктов и нашей практике внедрения.